Stereo-vision based Autonomous Underwater Navigation — the Platform SARSTION

Sheheryar Mehmood, Aadil Jaleel Choudhry, Hina Anwar, Saad Mahmood, Amir Ali Khan
NUST School of Electrical Engineering and Computer Science (SEECS), Islamabad, Pakistan

***Abstract*—Underwater robotics has been the center of attention for a long time due to diverse multi-disciplinary applications that it addresses. Conventionally, the detection and navigation has been governed by the SONAR based systems. While, SONAR offers unmatched capability for long range operation, its performance is compromised for detection of short range objects. In this regard, an alternate solution based on image processing appears promising in short range applications. Stereo-vision based systems are now regularly deployed for surface applications with huge success. However, the underwater deployment of these stereo-vision based systems is still an open problem. In the current work, we present an underwater autonomous platform, navigated by stereo-vision based algorithm. Initially, we demonstrate the distinct edge that the stereo-vision enjoys over SONAR in terms of its low cost, small size and low energy requirements. We present a working prototype of a Semi-Autonomous Sub-Aquatic Robot for Surveillance using Stereo-vision (SARSTION). SARSTION provides 6 degrees of freedom, using propellers, 4 for horizontal motion and 2 for vertical motion and is administered by an elaborate control system based on various positional feedback sensors. The main contribution of the current work is in the design of a novel simplistic navigational algorithm. The two acquired images from a front looking stereo-camera are used to compute a depth map whose intensity represents the relative distance of an object from the camera. The intensities are then equated to compute the relative physical distances of different objects from the camera. This information coupled with the approximations of relative object sizes and the data from the positional feedback sensors are then used to carefully alter the yaw of the mechanical assembly to maneuver the robot. The camera can also be used to provide live feed, depending upon its usage. We present some promising results illustrating the successful operation of the robot in a controlled reservoir.**

***Keywords*—underwater, autonomous robot, stereo-vision, navigation, AUV.**

I. Introduction

Autonomous Underwater Vehicles (AUVs) were introduced in the 1950s with the purpose of studying diffusion, acoustic transmission and submarine wakes. Remotely Operated Underwater Vehicles (ROVs) were developed in 1960s and one of the earliest were funded by US Navy with the objective of performing deep-sea rescue and object recovery operations. AUVs and ROVs can reach deeper waters than human divers and can submerge to shallower depth than manned vehicles. They are modular and scalable as per needs: allowing users to install additional equipment and sensors. Hence, they are an attractive and cost-effective option for naval defense, assistance in development of offshore oil fields, search-and-rescue operations and ocean-based research. They can be used to spy on hostile navy as well as for detection and destruction of underwater mines [1]. They can perform tasks like inspection of underwater cables and pipelines [2], [3]. Sensors can be installed on AUVs and ROVs to create ocean floor maps and study biodiversity [4].

Since World War II, when SONAR technology was being researched and developed at a rapid pace, to date, most underwater vehicles have employed the acoustic properties in detection and navigation [5], [6]. Despite being commonly used in all major naval projects as well as in minor AUVs and ROVs, it has limitations [7]. SONAR has low distance resolution and is suitable for large objects over longer distances. Underwater, a single transducer pair can't detect shape of objects. For that, an array is required, adding to complexity, cost, area, and power usage. Stereo-vision based systems are now regularly deployed for surface applications with huge success [8], [9]. Underwater stereo-vision was introduced about 15 years ago, focusing on object detection and 3D reconstruction [10], [11]. Stereo-vision has been developed for underwater topography and is currently being researched for identification and classification of marine life forms [12]. Stereo-vision offers higher distance resolution over short distances enabling detection of smaller objects. It also decreases cost, area and power consumption as compared to SONAR. Therefore, it is a novel technique being employed for navigation of AUVs [13]. However, luminosity, reflection and quality of water are among the factors which need to be catered for while employing stereo-vision effectively.

This paper presents an algorithm which builds upon stereo-vision, allowing navigation by real-time detection and avoidance of objects underwater. To verify the developed algorithm, an AUV was manufactured and put to use under test conditions.

II. Assembly Design

A mechanical assembly was required to test the working of the algorithm. For the purpose, a stainless steel hydrodynamic assembly was manufactured, with approx. dimensions of 35.56cm x 30.48cm x 15.24cm. The choice of stainless steel for the assembly was motivated by the team's lack of access to better manufacturing and machining facilities (like laser cutting, for instance). Steel is strong, malleable, and ductile; it can be welded and adequately sealed (using argon/TIC welding).

The steps followed in the manufacturing can be succinctly elucidated as follows: The 7.62cm diameter stainless steel pipes were stacked and welded in place as shown in Figure 1. They, together with the 3mm steel strips, formed the endoskeleton, providing support for the steel sheet covering. Moreover, these steel pipes function as the guiding channels for water propulsion. The motors were fitted inside these pipes: 2 motors for vertical motion and 4 for horizontal motion. Next,

978-1-4673-9127-6/16/$31.00 ©2016 IEEE

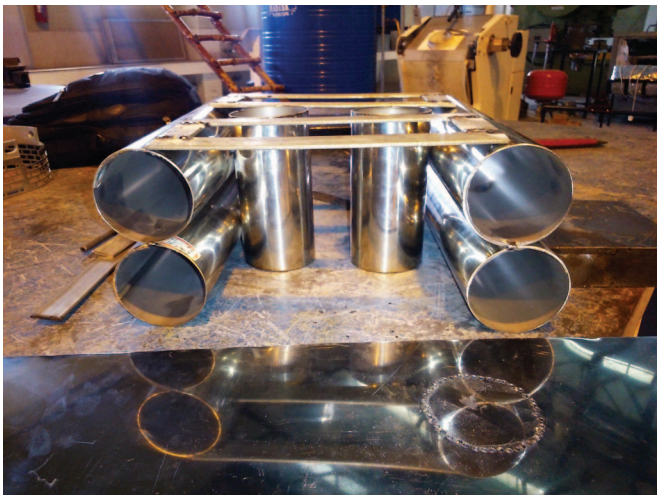


Fig. 1. Endoskeleton of robot

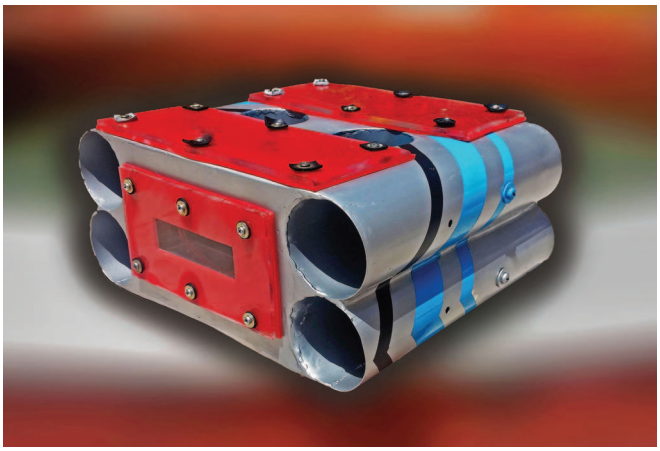


Fig. 2. Complete assembly of robot

a 1mm thick stainless steel sheet was wrapped along the length of the pipes as shown in the Figure 2. Prior to wrapping, circular and rectangular holes (in alignment with the steel pipes and component windows) were precisely cut in the sheet. The steel sheet was kept in one piece to minimize welding mistakes.

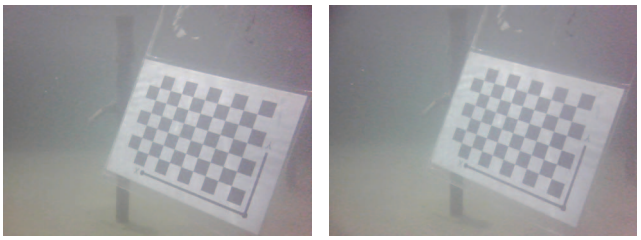
The assembly was made neutrally buoyant. The inertial measurement unit, along with batteries, on-board processor and controller, was placed inside, and the pressure sensor was mounted on top. The assembly had two openable windows at the top for component replacement. These windows were made waterproof using rubber gasket sealing and a preventive layer of silicone. The rubber sealing sheet was pressed between the top plates and the main steel chassis and was kept pressed using bolts. Table I summarizes physical aspects of assembly of robot.

III. Stereo-vision Algorithm

A stereo-camera was chosen which provides two 640 x 480 pixel horizontally concatenated images at 30 frames/second. The images are of same view from perspective separated horizontally by 5cm, called the baseline distance. OpenCV was used to interface the stereo-camera.

TABLE I. Physical aspects of assembly of robot

Property	Value
Height	15.24cm
Length	30.48cm
Width	35.56cm
Weight	9.9kg
Volume	9900cm^3



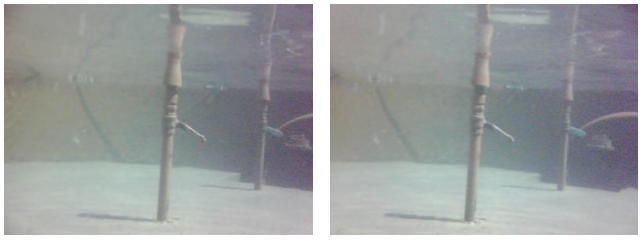
(a) Image from left camera (b) Image from right camera

Fig. 3. Images of calibration pattern from stereo-camera

One concatenated frame is grabbed and broken into left and right frames programmatically. Camera calibration is performed to obtain intrinsic and extrinsic parameters of camera [14], [15], [16]. For the purpose, a 9 x 6 checkerboard pattern is used. Attempt is made to minimize reprojection error to around 0.2%. Figure 3 shows images, taken from the stereo-camera, of the calibration pattern placed underwater.

To generate disparity maps, an image pair of desired view from stereo-camera is grabbed as shown in Figure 4 Rectification is done using camera parameters to remove distortions from image and make them appear as though the two images are parallel. Block Matching Algorithm is used to compute disparity maps. Disparity map obtained from images in Figure 4 is shown in Figure 5. Parameters used to tune the algorithm are given in Table II. Test objects are placed in various positions and their corresponding gray-scale intensities are noted. These values represent relative distance of objects from one another. These gray-scale intensities along with real distance of test objects are interpolated to fit a polynomial to map any gray-scale value on real distance. Same procedure is followed to map dimensions of image to height and width of the view. Figure 6 represents both mappings graphically.

Disparity maps can be noisy. This happens due to difference in focal lengths of cameras, low luminosity and reflections. If these factors cannot be catered for in hardware, software filtering is done. Noise is suppressed by first applying a temporal median filter to 4 successive disparity maps of same view. The resultant frame shown in Figure 7 is subjected to thresholding at different gray-levels to extract objects at corresponding distances. An example is shown in Figure 8. Each of the resulting binary image is eroded by a relatively small structuring element. Optimum results were obtained by thresholding at 60.96cm to 121.92cm at every 15.24cm and by using a 30 x 10 structuring element. 60.96cm and 121.92cm are referred to as minimum algorithmic distance and maximum algorithmic distance respectively. 15.24cm is referred to as thresholding distance. Hit-miss transformation is applied to each resultant binary image with a structuring element equivalent to the shape and size of robot at that



(a) Image from left camera (b) Image from right camera

Fig. 4. Images of test environment from stereo-camera

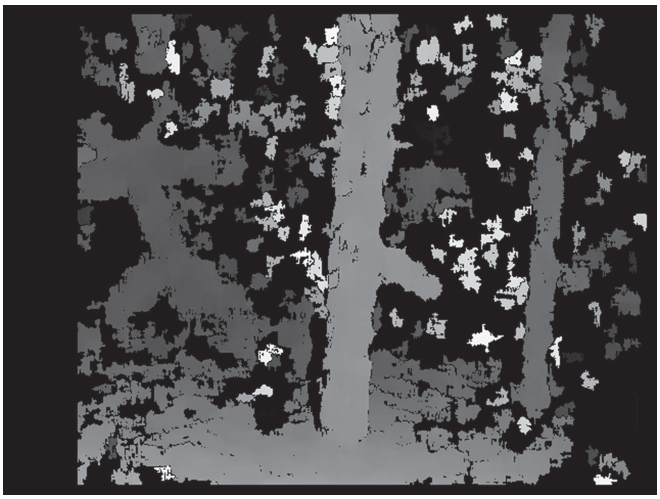


Fig. 5. Disparity map generated from images in Figure 4

distance. All images are then combined to give a *path* image as shown in Figure 9. White pixels represent free path whereas black pixels represent obstructions. This image is analyzed to see if middle row of the image has any white pixel in presence of which, the yaw of robot is set as to bring the white pixel to center of image. If there is no white pixel, robot is set to turn until one such pixel appears. Yaw is calculated using simple trigonometry. Same can be extended to depth of robot by analyzing the central column of the image.

IV. Mathematical Modeling of Navigational Algorithm

Disparity maps represent 3 dimensional model as 2 dimensional image. In Figure 10a, stereo-camera is represented by point *C* at a distance *d* from image plane. This plane is characterized by an intensity level *i*, a height span[1] of *hs* and width span of *ws*. The distance of camera from floor or water bed is *dp* and *n* represents the noise pixels from floor in that image plane [17].

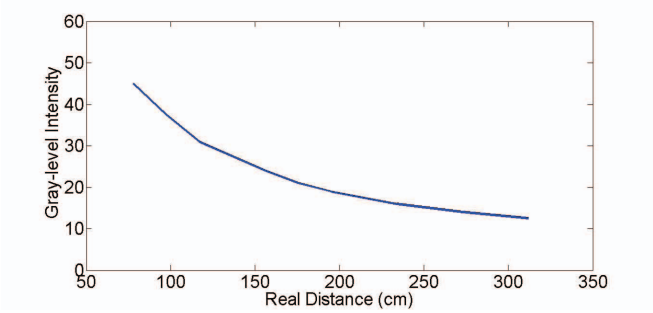
Interdependence of these parameters can be given by:

$$i = I(d) \tag{1}$$
$$ws = Ws(d) \tag{2}$$
$$hs = Hs(d) \tag{3}$$
$$n = N(d, dp) \tag{4}$$

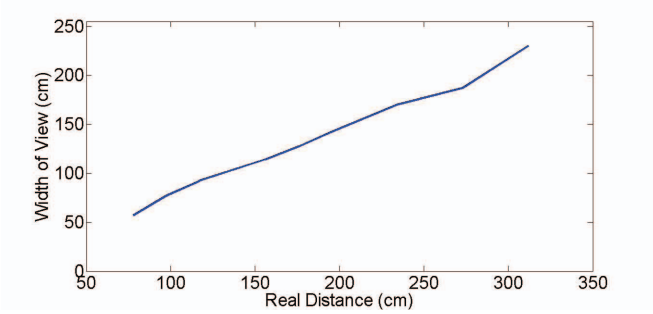
[1]Height span is the distance equivalent to the total image height or vertical pixels. For camera described in Section III, it is 480.

TABLE II. Parameters used in Block Matching Algorithm

Parameters	Value
SAD window size	21
No. of Disparities	64
Texture threshold	10
Uniqueness Ratio	2
Pre-filter cap size	12



(a) Graph between real distance and corresponding gray-level intensity



(b) Graph between real distance and corresponding width of view

Fig. 6. Graph representing relationships between gray-level intensity in disparity map, real distance and width of view in image frame

Only *i* and *dp* are known. However, *i* is only dependent on *d* which can be found out as following:

$$d = I^{-1}(i) \tag{5}$$

It can be used to find the remaining parameters. In order to find *hs*, *ws* and *n*, Figure 10a has been redrawn as Figure 10b from which it can be shown that:

$$tan(\frac{\theta hs}{2}) = \frac{hs/2}{d} \tag{6}$$

Simplifying Equation 6 gives:

$$hs = (2 * tan(\frac{\theta hs}{2})) * d \tag{7}$$

Similarly:

$$ws = (2 * tan(\frac{\theta ws}{2})) * d \tag{8}$$

For *n*:

$$n = (\frac{hs}{2} - dp) * factor \tag{9}$$

In Equation 9, *factor* is used to convert distance in centimeters to number of pixels. For selected camera, it is given by:

$$factor = \frac{480}{hs} \tag{10}$$

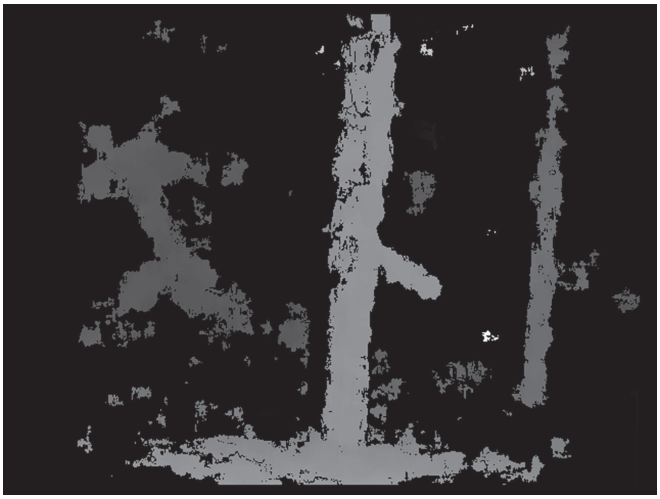


Fig. 7. Disparity map after applying spatial and temporal filters

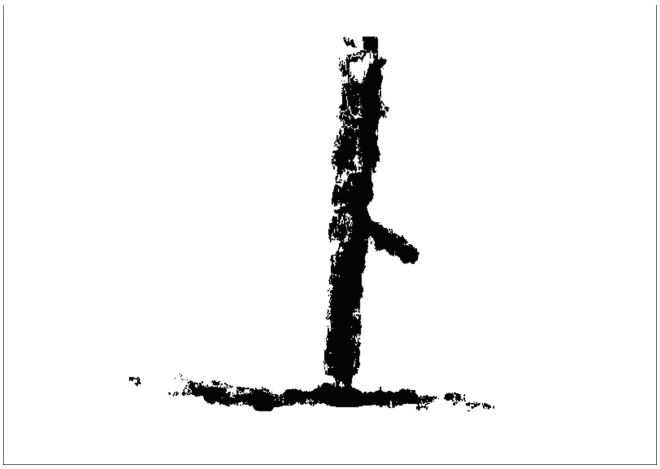


Fig. 8. Binary image representing the extracted object at a certain distance from camera

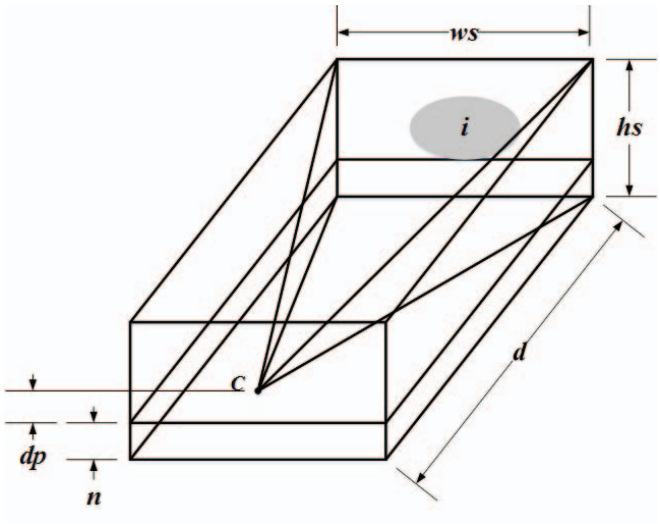
Using Equations 9 and 10, n becomes:

$$n = (\frac{1}{2} - \frac{dp}{hs}) * 480 \tag{11}$$

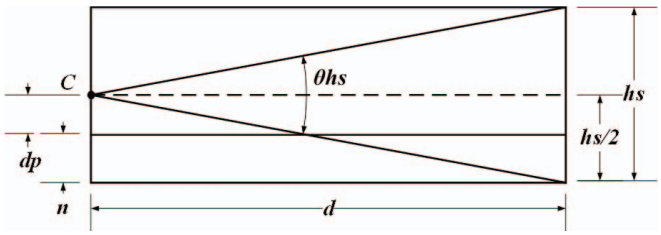
As all unknown parameters have been determined, navigational algorithm can be summarized as following:

- Acquire current yaw of robot from Control System Unit (CSU) on SARSTION.
- Grab image pair from stereo-camera.
- Generate disparity map and suppress noise as described in Section III.
- Set current distance d to minimum algorithmic distance.
- Set the path image to all ones i.e., white pixels.
- While current distance has not exceeded maximum algorithmic distance:
  - Find gray-level intensity I in disparity map at that distance.
  - Extract binary image plane at that distance by thresholding the gray-level intensity $I \pm step$[2].
  - Apply hit-miss transformation as described in Section III.
  - AND the path image with resulting image and store.
  - Increment current distance d by thresholding

Fig. 9. Path image representing free path as white pixels and obstructions as black pixels



(a) Orthographic view



(b) Side view

Fig. 10. Mathematical modeling of stereo-camera

[2]*Step* is change in gray-level intensity per thresholding distance.

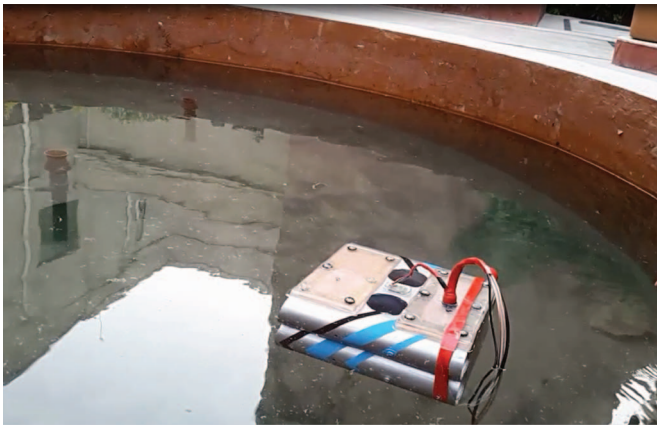


Fig. 11. Testing of SARSTION in reservoir

TABLE III. PROCESSING UNIT SPECIFICATIONS

Processor	Intel Core i5-2430M CPU @ 2.40Ghz
RAM	4GB
Operating System	Windows 8.1 Pro 64-bit
Programming Language	C/C++

distance *D* which is equivalent to *step*.
- Repeat while loop.

- Check for a white pixel in center of path image. If found, signal CSU to retain current yaw.
- If pixel in center of path image is not white:
  - Find string of white pixels at left side from the center of path image. Find mean white pixel from first and last white pixel in string. Find mean distance, the distance between mean pixel and center pixel in path image.
  - If left side has no white pixel, find string of white pixels at right side from the center of path image. Find mean white pixel from first and last white pixel in string. Find mean distance, the distance between mean pixel and center pixel in path image.
  - If right side has no white pixels, signal CSU to change yaw or robot till such pixels appear.
- Find corresponding angles from center using maximum algorithmic distance, height span and mean distance.
- Signal CSU to change yaw of robot accordingly.

V. RESULTS AND DISCUSSION

SARSTION was manufactured at a cost of $600. It was put to test in a controlled cylindrical reservoir of diameter 3.65m and depth 1.22m under naturally lit conditions as shown in Figure 11. Trials were carried out for a period of over a month under different lighting conditions and states of water from clear to semi-turbid. Multiple obstacles of different shapes and sizes under different configurations were placed. Algorithm was run on processing unit described in Table III.

Processing unit was able to generate up to 14 path frames per second, making the algorithm real-time. The algorithm was able to detect objects clearly at minimum distance of up to 60.96cm and maximum distance of up to 304.8cm. The algorithm was able to distinguish objects as small as 2cm x 2cm at a precision of 10cm along depth and 2cm along width and height in this range. Operating at maximum speed of 2m/s, SARSTION was able to successfully navigate through 80% of test cases without collision.

Failures can be attributed to lack of sufficient light, limited resolution and small baseline distance of camera and weak noise suppression techniques. Mounting a costlier camera with higher resolution and larger baseline distance increase precision as well as decrease minimum distance of object detection. This, along with machine learning algorithms to adapt to noise can increase results by many folds. However, both of these require faster processing unit which results in further increase in cost. A cheaper solution can be to mount a blue light source on SARSTION which can decrease noise as well as increase maximum distance of object detection and compensate for weak noise suppression algorithm.

VI. CONCLUSION

SARSTION is a low-cost AUV navigated by a simplistic real-time stereo-vision based algorithm. Results indicate that stereo-vision can be used to generate signals to maneuver an AUV through an array of obstacles. The prototype, though adequately functional, is still somewhat crude. It has a considerable room for improvement, especially in regards to control system optimization and adaptation to the varying environmental factors. More accurate path planning techniques can be employed to further reinforce the viability of this concept. The applications of AUVs navigated by stereo-vision are diverse: ranging from defense purposes to oil exploration and research orientation towards bio-diversity. Given a basic prototype, the AUV can be modified in form and function to meet the varying requirements of different applications. For instance, additional equipment, like mechanical arms, can be added to the rudimentary design. Commercialization of this concept can stimulate innumerable benefits, and can encourage the much needed research in this evolving and highly promising domain.

ACKNOWLEDGMENT

The authors would like to thank Dr. Ammar Hassan, Assistant Professor NUST SEECS, for his guidance in developing CSU of SARSTION. The authors would also like to thank Dr. Arshad Ali, Ex. Principal NUST SEECS, Mr. Syed Ali Abbas, Centres of Excellence in Science & Applied Technologies, and Dr. Faisal Shafait, Assistant Professor NUST SEECS, for their support.

REFERENCES

[1] D. P. Williams, "On optimal auv track-spacing for underwater mine detection," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*. IEEE, 2010, pp. 4755–4762.

[2] J. Nicholson and A. Healey, "The present state of autonomous underwater vehicle (auv) applications and technologies," *Marine Technology Society Journal*, vol. 42, no. 1, pp. 44–51, 2008.

[3] Y. Petillot, S. Reed, and J. Bell, "Real time auv pipeline detection and tracking using side scan sonar and multi-beam echo-sounder," in *OCEANS'02 MTS/IEEE*, vol. 1. IEEE, 2002, pp. 217–222.

[4] L. A. Mayer, "Frontiers in seafloor mapping and visualization," *Marine Geophysical Researches*, vol. 27, no. 1, pp. 7–17, 2006.

[5] L. Stutters, H. Liu, C. Tiltman, and D. J. Brown, "Navigation technologies for autonomous underwater vehicles," *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, vol. 38, no. 4, pp. 581–589, 2008.

[6] A. Elfes, "Sonar-based real-world mapping and navigation," *Robotics and Automation, IEEE Journal of*, vol. 3, no. 3, pp. 249–265, 1987.

[7] S. B. Williams, P. Newman, G. Dissanayake, and H. Durrant-Whyte, "Autonomous underwater simultaneous localisation and map building," in *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, vol. 2. IEEE, 2000, pp. 1793–1798.

[8] A. H. A. Hasan, R. A. Hamzah, and M. H. Johar, "Region of interest in disparity mapping for navigation of stereo vision autonomous guided vehicle," in *Computer Technology and Development, 2009. ICCTD'09. International Conference on*, vol. 1. IEEE, 2009, pp. 98–102.

[9] F. Rovira, Q. Z. Más, and J. Reid, "Automated agricultural equipment navigation using stereo disparity images," 2004.

[10] S. Negahdaripour and P. Firoozfam, "An rov stereovision system for ship-hull inspection," *Oceanic Engineering, IEEE Journal of*, vol. 31, no. 3, pp. 551–564, 2006.

[11] V. Brandou, A.-G. Allais, M. Perrier, E. Malis, P. Rives, J. Sarrazin, and P.-M. Sarradin, "3d reconstruction of natural underwater scenes using the stereovision system iris," in *OCEANS 2007-Europe*. Ieee, 2007, pp. 1–6.

[12] M.-C. Chuang, J.-N. Hwang, K. Williams, and R. Towler, "Tracking live fish from low-contrast and low-frame-rate stereo videos," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 25, no. 1, pp. 167–179, 2015.

[13] P. Negre, F. Bonin-Font, and G. Oliver, "Stereo graph slam for autonomous underwater vehicles," in *Proceedings of the The 13th International Conference on Intelligent Autonomous Systems, Padova, Italy*, 2014, pp. 15–19.

[14] C. Guodong, M. Xie, Z. Xia, L. Sun, J. Ji, Z. Du, and W. Lei, "Fast and accurate humanoid robot navigation guided by stereovision," in *Mechatronics and Automation, 2009. ICMA 2009. International Conference on*. IEEE, 2009, pp. 1910–1915.

[15] L. M. Song, M. P. Wang, L. Lu, and H. J. Huan, "High precision camera calibration in vision measurement," *Optics & Laser Technology*, vol. 39, no. 7, pp. 1413–1420, 2007.

[16] R. Y. Tsai, "A versatile camera calibration technique for high-accuracy 3d machine vision metrology using off-the-shelf tv cameras and lenses," *Robotics and Automation, IEEE Journal of*, vol. 3, no. 4, pp. 323–344, 1987.

[17] D. Rosselot and E. L. Hall, "The xh-map algorithm: A method to process stereo video to produce a real-time obstacle map," in *Optics East 2005*. International Society for Optics and Photonics, 2005, pp. 60 060K–60 060K.